Energy-aware scheduling in distributed computing systems

Santiago Iturriaga

Directores: Sergio Nesmachnow y Bernabé Dorronsoro

Tesis de Doctorado en Informática

PEDECIBA Informática
Universidad de la República
Uruguay

Index

Motivation

- Distributed computing systems
  - Key for supporting modern computing demands
  - Provide services for science, industry, and commerce
- Energy consumption has become a major concern
  - 4.5% annual increase worldwide (Andrae and Edler, 2015)
- Optimizing energy efficiency is challenging
  - Conflicts with optimizing performance and QoS

Towards energy efficiency in data centers

- Most effective energy efficiency approaches:
  - Optimizing computing components
  - Optimizing cooling components
  - Considering renewable energy sources
- Scheduling the operative of data centers is key but challenging
  - General scheduling problem is NP-hard
  - Multiple conflicting objectives
  - Uncertainty in scenarios
- Uncertainty in scenarios: state of the art approaches
  - Fail to simultaneously consider user estimates, multicore, and energy
  - e.g., Tang et al. (2013), Yu et al. (2015), Chen et al. (2016c)

Towards energy efficiency in data centers

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Many works do not consider a multiobjective (MO) approach
  - e.g. Goudarzi and Pedram (2016), Shi et al. (2017), Lee et al. (2017)
- Single data center: state of the art MO scheduling
  - Fail to simultaneously consider QoS, cooling, and renewable energy
  - e.g., Lei et al. (2015), Tang et al. (2016), Xie et al. (2016)
- Federation of data centers: state of the art MO scheduling
  - Fail to simultaneously consider QoS, multicore architectures, and multiple precedence-constrained jobs
  - e.g., Jena (2015), Habibi Khalaj et al.(2015), Kaushik and Vidyarthi (2016)

Goal: address scheduling of energy efficient data centers

- Accurate model for single and federation of data centers
- Multiobjective approach
- Study the impact of uncertainty

Index

1. Introduction
2. Scheduling a single energy-efficient data center
3. Scheduling a federation of energy-efficient data centers
4. Robustness of energy-aware schedulers
5. Conclusions and future work

Scheduling a single energy-efficient data center

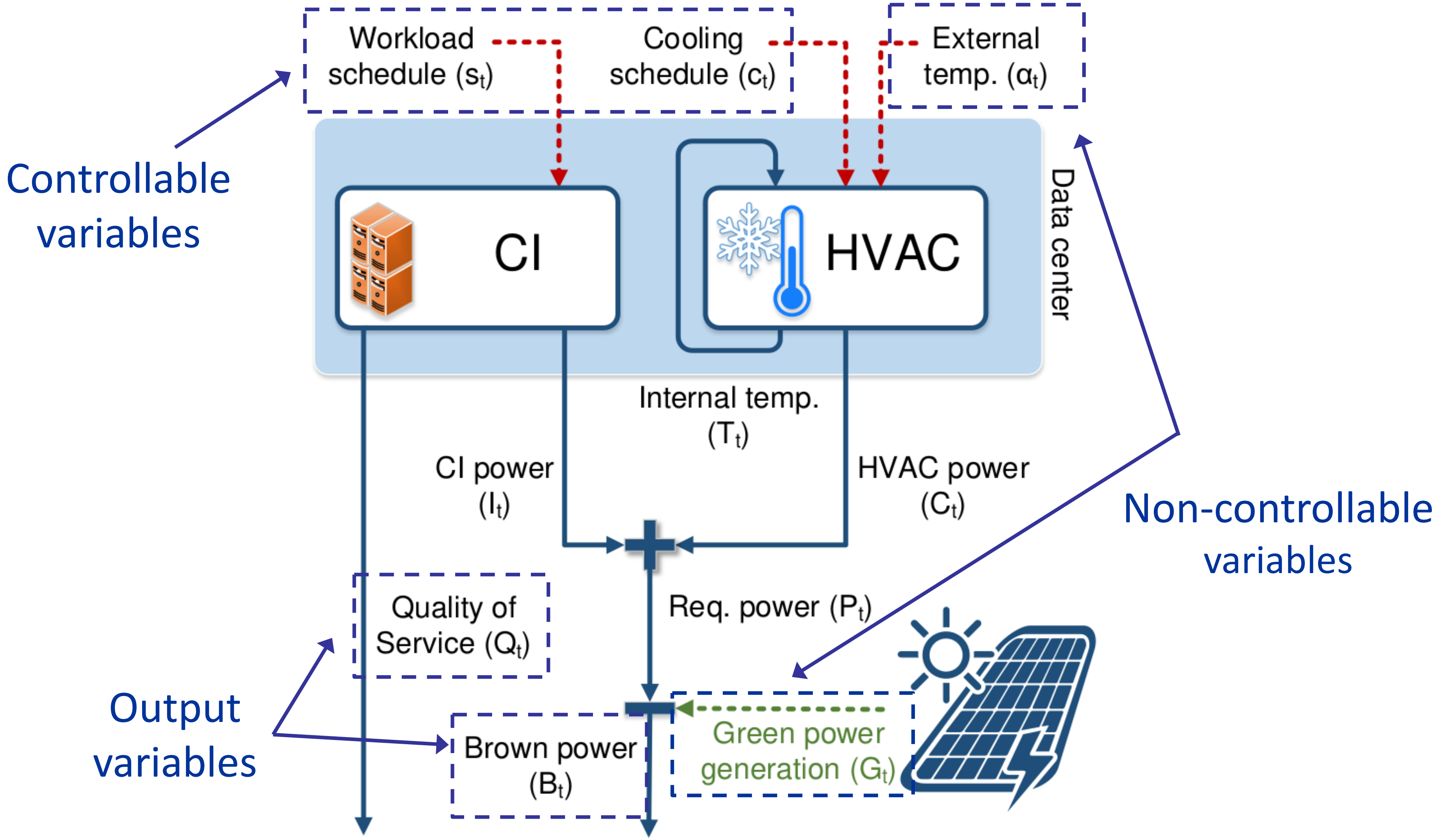
Overview

- Schedule the operation of a single data center
- Controlling computing and cooling components
  - Low-powered servers and free cooling
- Independent tasks with due dates
- Hybrid energy sources: traditional and solar
- Ancillary services for energy power
  - Deviation from reference power profile
- Subject to maximum room temperature

Iturriaga, S. and Nesmachnow, S. (2016). Scheduling energy efficient data centers using renewable energy. Electronics, 5(4):1-16

Scheduling a single energy-efficient data center

Data center model



Optimization objectives

$$min\ z_p = \sum_{t=1}^{K} \begin{cases} (P_t - R_t)/\max(R_t), & \text{if } P_t > R_t, \\ 0, & \text{if } P_t \leq R_t, \end{cases}$$

reference profile

$$min\ z_b = \sum_{t=1}^{K} B_t \times M_t^b,$$

energy budget

$$min\ z_q = \sum_{i=1}^{N} \begin{cases} FT(i) - D(i), & \text{if } FT(i) > D(i), \\ 0, & \text{if } FT(i) \leq D(i). \end{cases}$$

QoS

subject to:

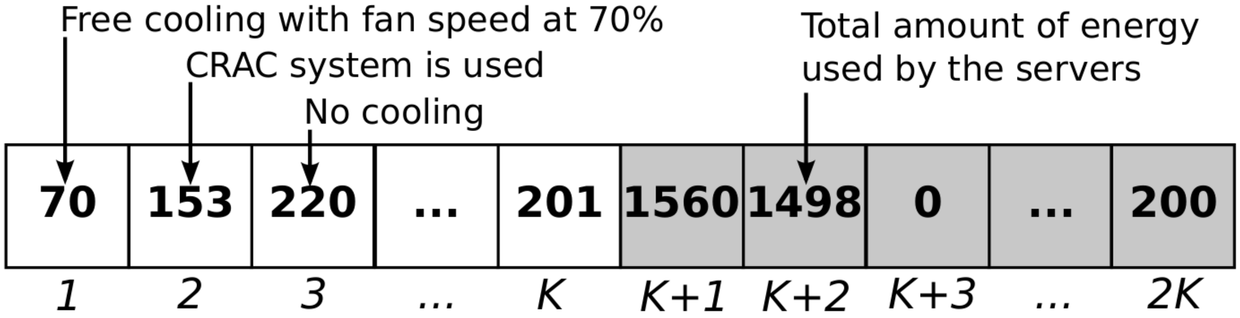
$$T_t \leq \hat{T}, \qquad t = 1 \ldots K$$

temperature contraint

- R_t: ref. power profile, $\hat{T}$: max. temperature, M_t^b: brown energy cost, $D(i)$: due date and $FT(i)$: finishing time of task i

Proposed algorithms

- Multiobjective evolutionary algorithm: NSGA-II and ev-MOGA
  - Schedule servers power states and cooling components
  - Energy consumption configuration

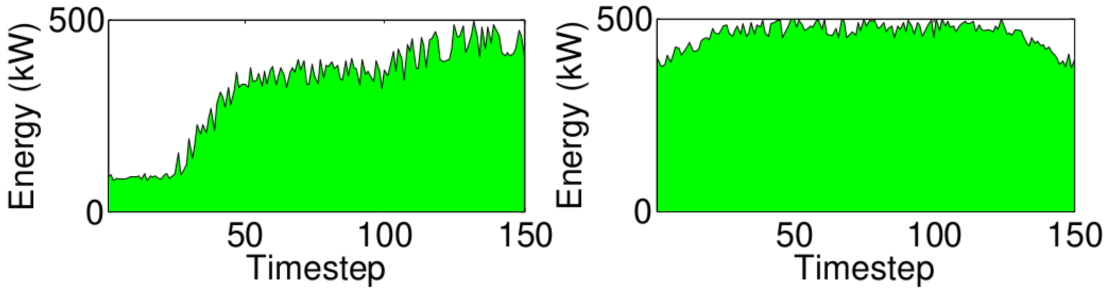


- Strong hybridization
  - Task scheduling: Best Fit Hole (BFH) greedy heuristic
  - Keeps track when servers are idle (holes)
  - Assigns a task to the hole where it best fits
- Weak hybridization: post hoc optimization
  - Task scheduling: Simulated Annealing (SA)
  - Applies a simple task moving operation
  - Dominance is used as acceptance criterion

Problem instances

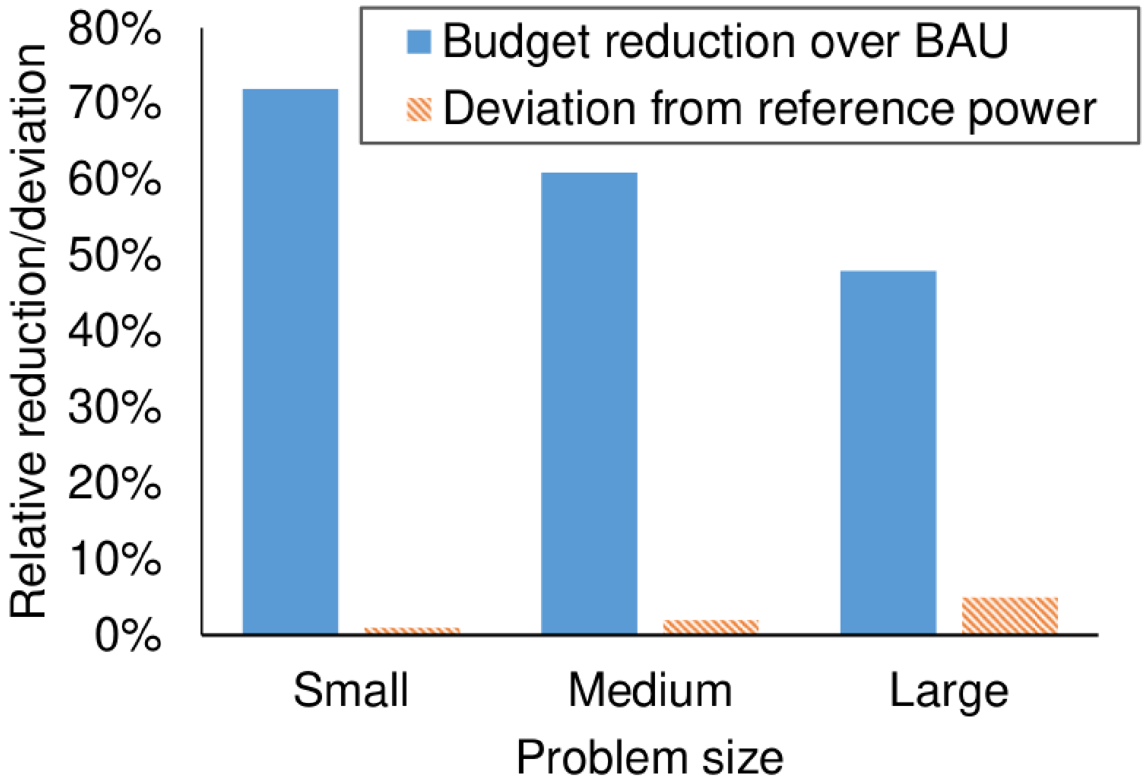
UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Scheduling horizon: 150 minutes
- Workloads: 200, 300, 400 tasks
- Computing Infrastructure
  - 64 low-power Intel Atom servers: 30 W max, 22 W idle, 3 W sleep
- HVAC energy consumption
  - CRAC consume 2.3 kW and fans between 0 to 410 W
- Traditional brown energy pricing scheme
  - low, medium (2x), high (4x) profiles
- Photovoltaic generator of 1.5 kW
  - morning (*g1*), midday (*g2*), night (*g3*) profiles



Experimental results

- ev-MOGA significantly outperforms NSGA-II
- ev-MOGA (due dates met ≥ 95%): improv. over BAU
  - Business As Usual (BAU): servers never *sleeps* and no *green energy*



Scheduling a single energy-efficient data center

Summary of main contributions

- Model of a modern data center powered by hybrid energy
- Schedule server states, cooling devices and workload of tasks
- Multiobjective planning: energy budget, reference power profile and due dates met
- Accurate evolutionary algorithms for solving the problem
- Comparing with business as usual
  - reduce budget between 33%-83% with
  - less than 3% deviation from ref. profile and
  - more than 95% due dates met

Index

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Overview

- Federation: a set of data centers cooperating with each other
- Schedule the operation of a federation of data centers
  - Geographically distributed data centers
  - Multicore computing components
  - Workflows of parallel tasks with deadlines
- Optimization objectives
  - Minimize: makespan, energy consumption, violations to SLA

Iturriaga, S., Dorronsoro, B., and Nesmachnow, S. (2017). Multiobjective evolutionary algorithms for energy and service level scheduling in a federation of distributed datacenters. International Transactions in Operational Research, 24(1-2):199-228

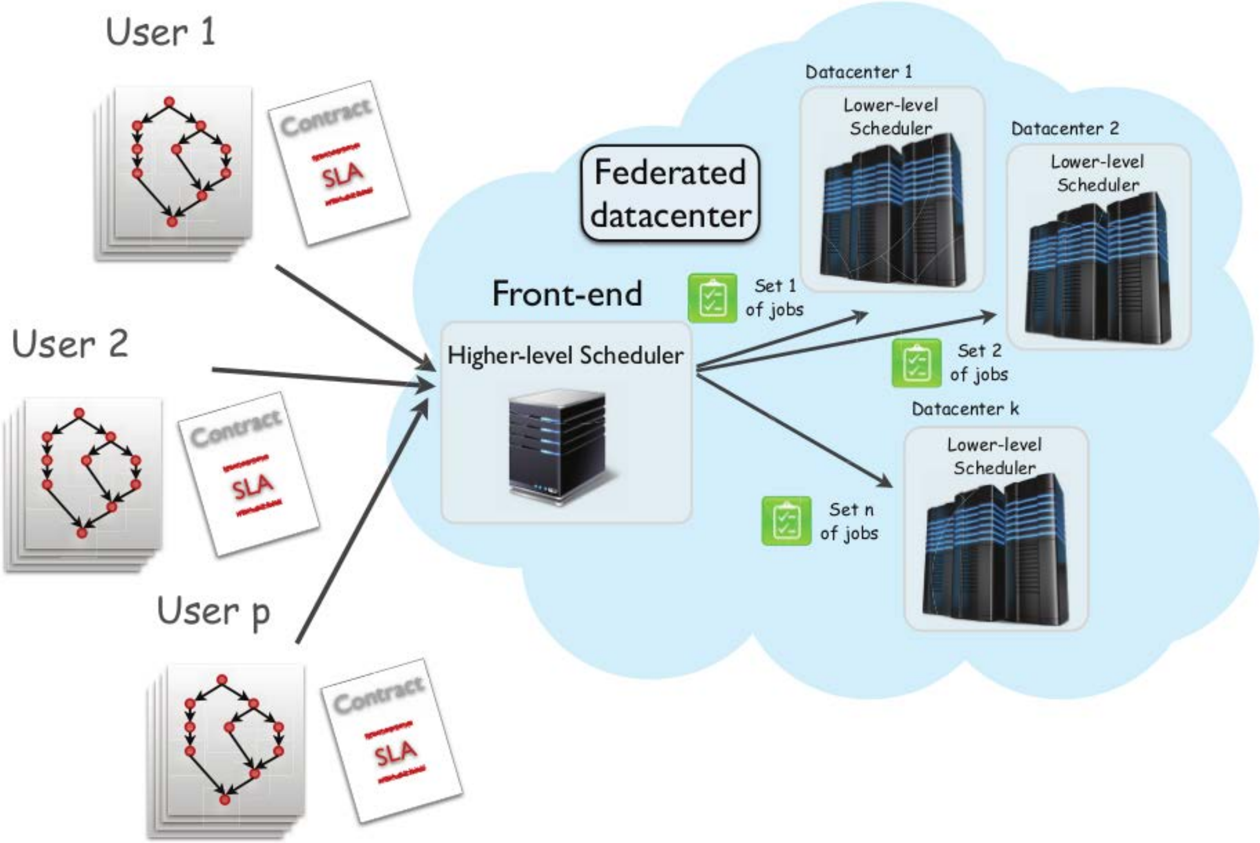
Federation of homogeneous data centers

Problem model

- A set of *homogeneous* datacenters, $CN = \{CN_1, \dots, CN_k\}$
  - np_j: number of processors, c_j: number of cores of each processor, ops_j: performance (FLOPS), $(e_j^{idle}, \ e_j^{max})$: energy consumption at idle/max
- A set of heterogeneous jobs $J = \{j_1, \dots, j_n\}$ with deadline d_q
  - Comprised of a (large) set of tasks $WT_q = \{wt_1, \dots, wt_m\}$ with dependencies
  - Each task wt_q is defined by o_q: number of operations, and nc_q: number of cores required
- Each user owns a set of jobs
  - SLA determines the percentage of jobs that must meet their deadlines
- Communication costs negligible between servers within the same CN

Federation of homogeneous data centers

Problem model



- Two-level scheduling model
  - *Higher-level scheduler*: schedules jobs to data centers
  - *Lower-level scheduler*: schedules tasks within each datacenter

Federation of homogeneous data centers

Optimization formulation

- Minimize:

$$f_M(\vec{x}) = \max_{0 \leq r \leq k} CT_r$$

makespan

$$f_E(\vec{x}) = \sum_{r \in DC} \sum_{\substack{q \in Q: \\ f_1(q)=r}} \sum_{\substack{wt_\alpha \in WT_q: \\ f_2(wt_\alpha)=s_j}} \frac{o(wt_\alpha)}{ops(s_j)} \times e_{s_j}^{max} + \sum_{s_j \in S_r} e_{s_j}^{idle}$$

energy consumption

$$f_S(\vec{x}) = \sum_{u_i \in U} \max\left(0, \left[\sum_{q \in wu(u_i)} Violated(q) - (1 - SLA_{u_i}) \times WF(u_i)\right]\right)$$

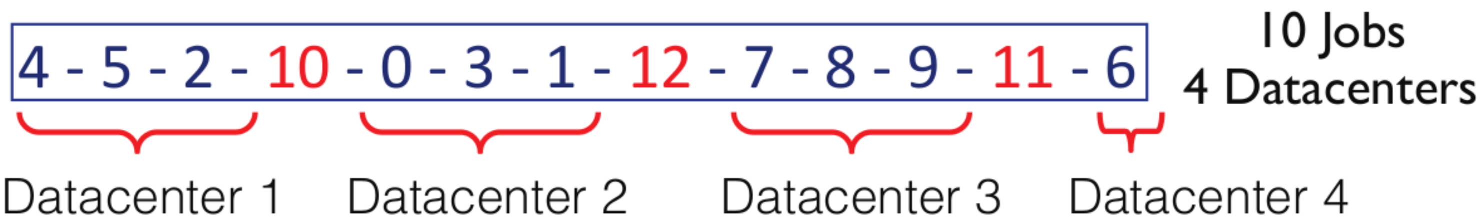
QoS

- $\vec{x}$ is a schedule and CT_r is the completion time of CN_r
- f_1: higher-level scheduling function; f_2: lower-level scheduling function
- $Violated(q) = 1$ if deadline of J_q is not met, $WF(u_i)$: num. of jobs of u_i

Federation of homogeneous data centers

Higher-level and lower-level schedulers

- Higher-level schedulers
  - Multiobjective evolutionary approaches: NSGA-II, MOCellSRF



  - Greedy heuristic approaches: Round robin, load balancing, MaxMin, MaxMIN, MinMIN
- Lower-level scheduler: Earliest Finishing Time Hole (EFTH)
  - Based on Heterogeneous Earliest Finish Time
  - Backfilling for multi-cores, considering holes for partial processor usage
  - Assigns tasks according to *upward rank* prioritizing the usage of holes

Federation of homogeneous data centers

Problem instances

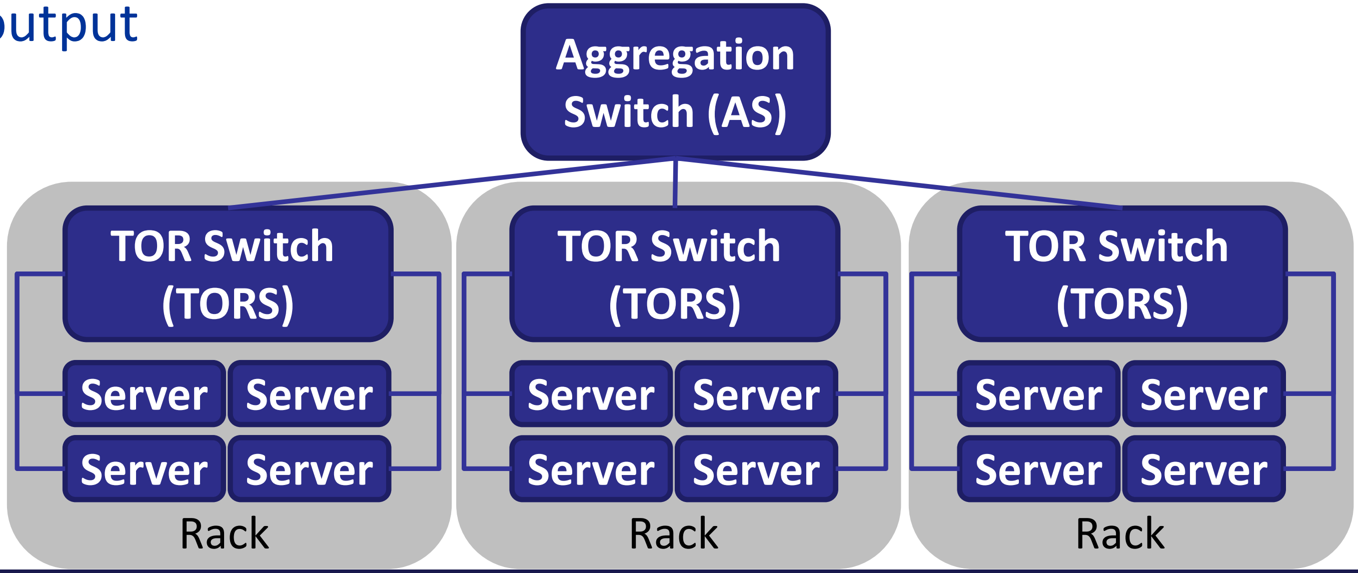
- Medium: 9 batches with 10 to 250 jobs (600 tasks each batch)
- Large: 125 batches with 1000 jobs (3 to 132 tasks each job)
- Federation of 5 CNs (up to 100 processors each)
- Workflow types
  1. *Series-Parallel*: concurrent threads running in parallel
  2. *Heterogeneous-Parallel*: non-identical tasks with arbitrary precedence
  3. *Homogeneous-Parallel*: identical tasks with arbitrary precedence
  4. *Single-Task*: only one task per job
  5. *Mixed*: 30% of types 1 to 3, 10% of type 4
- Three SLA levels: 98%, 94%, and 90%

Experimental results

- Medium instances
  - Constraint programming for computing lower bounds (LB)
  - GAP: relative difference between LB and computed value
  - *Makespan*: Series-Parallel and Heterogeneous-Parallel are the hardest to optimize with average GAP_M of 37%
  - *Energy consumption*: accurate schedules with GAP_E as low as 8%
  - *QoS*: all algorithms compute the lower bound value
- Large instances
  - MaxMIN: most accurate higher-level greedy heuristic
  - NSGA-II: most accurate *hypervolume* results
  - MOCell: most accurate *spread* results

Heterogeneous problem model

- A set of *heterogeneous* datacenters, $CN = \{CN_1, \ldots, CN_k\}$
  - Comprised of a set of racks, each with homogeneous processors
  - Network: intra-rack TORS speed rs_j, inter-rack AS speed as_j
  - Communication costs are negligible for tasks in the same processor
- A set of heterogeneous jobs $J = \{j_1, \ldots, j_n\}$ represented by a DAG
  - Each task wt_q considers dt_q, the normalized time required for transferring its output



Federation of heterogeneous data centers

Higher-level schedulers

- Multiobjective evolutionary algorithms: NSGA-II, IBEA, SMS-EMOA
- Greedy heuristics
  - CA-MaxMin: sorted descending by the product of execution time and the sum of cores required by all tasks and assigned minimizing completion time
  - Longest First: sorted descending the product of execution time, the sum of the execution time of all tasks, and the sum of cores required by all tasks
  - Job re-sorting algorithm is applied after heuristics
- Earliest Finishing Time Hole (EFTH) for lower-level scheduling

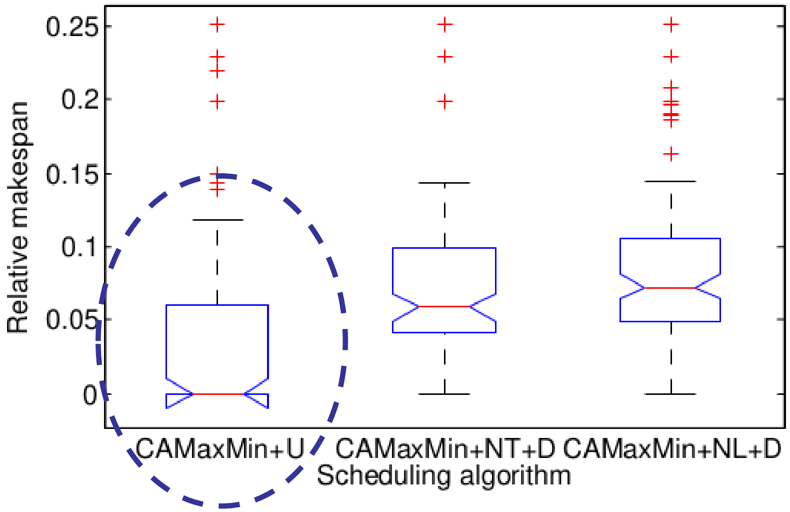
Federation of heterogeneous data centers

Problem instances

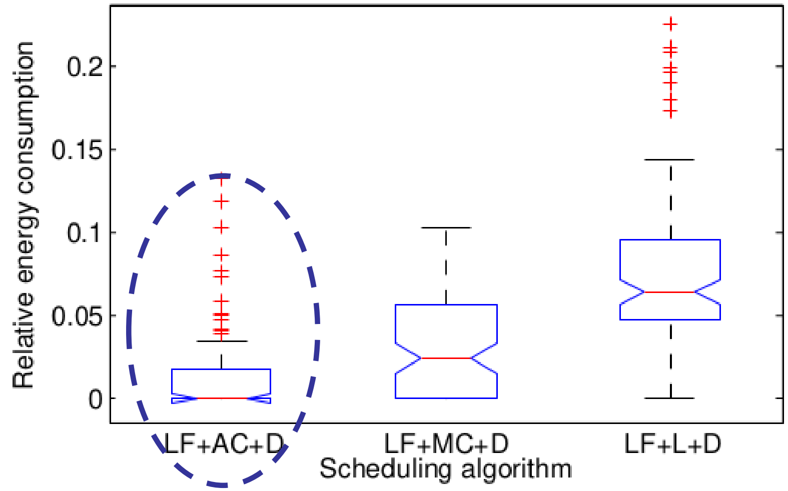
- 100 batches with 1000 jobs (3 to 132 tasks each job)
- Federation of 5 CNs
  - Small: average of 100 processors each
  - Medium: average of 325 processors each
- Communication time is 5-50% of the task computation time
- Racks: 18-42 processors networked by 1GbE or 10GbE
- Workflow types: Series-Parallel, Heterogeneous-Parallel, Homogeneous-Parallel, Single-Task and Mixed
- Three SLA levels: 98%, 94%, 90%

Federation of heterogeneous data centers

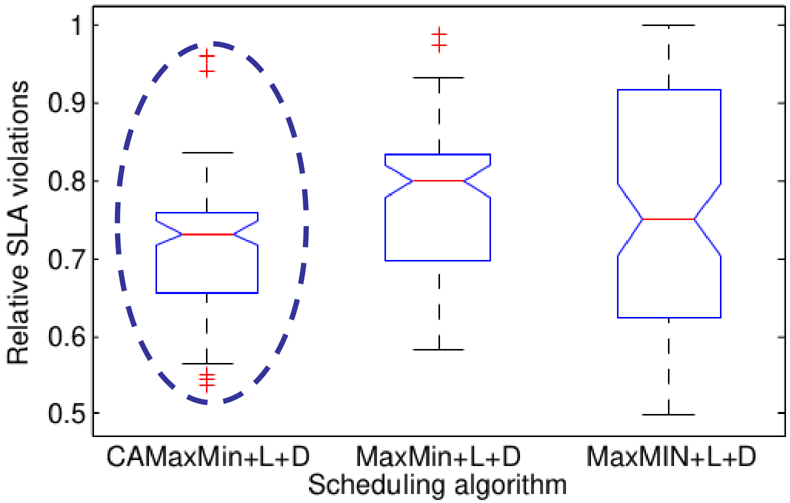
Experimental results: greedy heuristics



(a) Relative makespan

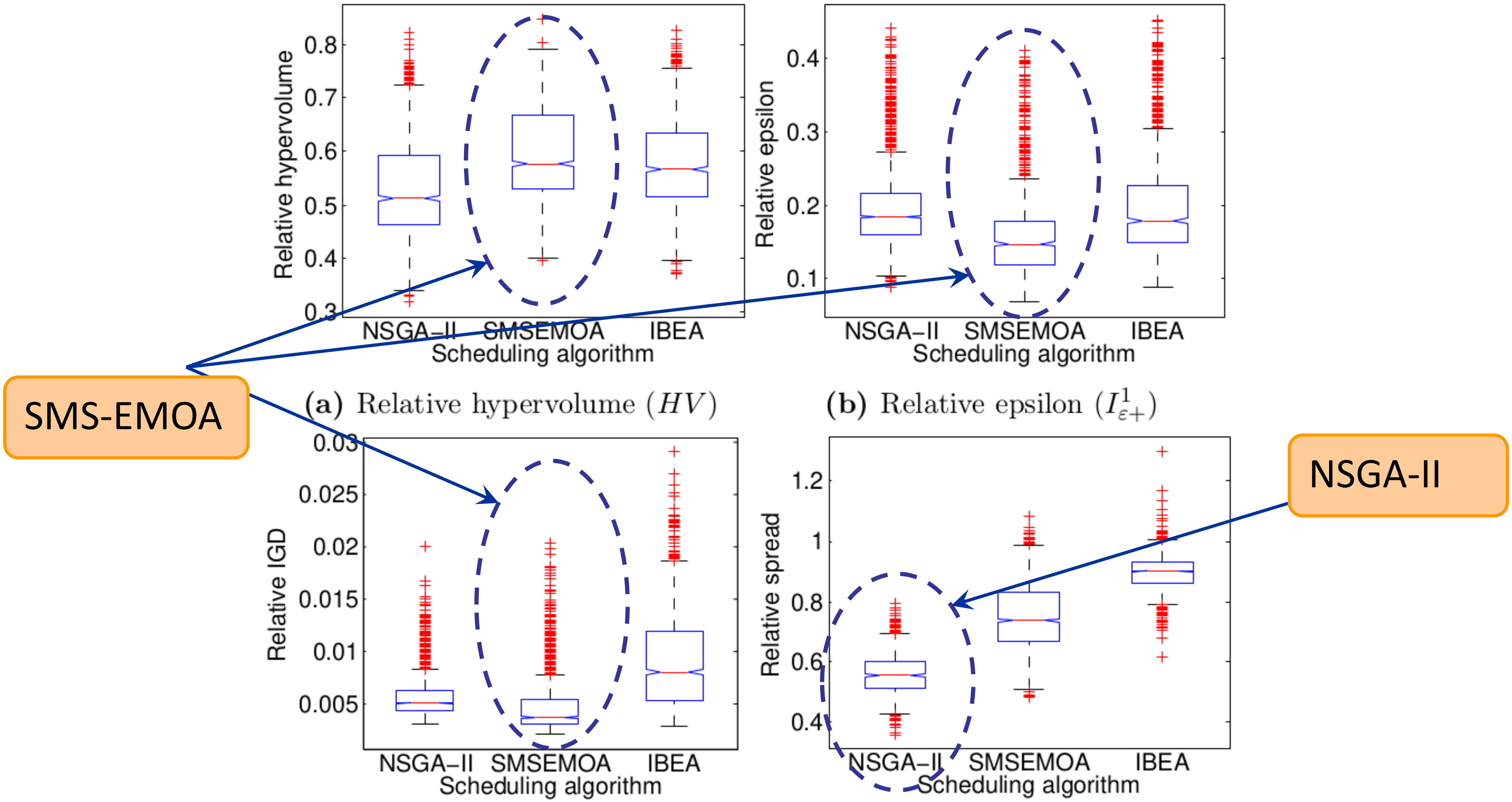


(b) Relative energy consumption



Federation of heterogeneous data centers

Experimental results: MOEAs



(a) Relative hypervolume (HV) **(b)** Relative epsilon ($I^1_{\varepsilon+}$)

(c) Relative IGD **(d)** Relative spread (Δ)

Summary of main contributions

- Multiobjective formulation for scheduling of a federation of data centers
  - Minimize: makespan, energy consumption, and SLA violation
  - Homogeneous and heterogeneous data centers
- Accurate hierarchical energy-aware approach
  - Online and offline algorithms
- A set of problem instances is proposed
- MaxMIN, CA-MaxMin and LF: most accurate higher-level heuristics
- SMS-EMOA: most accurate higher-level MOEA

Index

Robustness of energy-aware schedulers

Overview

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Execution time and energy consumption of tasks are uncertain
- Arguably a major factor of uncertainty is introduced by users
  - Users must specify the Estimated Execution Time (EET) of a task
- EET is highly inaccurate, with accuracy as low as 10%
  - Tasks fail because of initialization errors
  - Users overestimate EET to prevent task from being killed
- Empirical evaluation of energy-aware schedulers for data centers with uncertain execution time and energy consumption

Iturriaga, S., García, S., and Nesmachnow, S. (2014). An empirical study of the robustness of energy-aware schedulers for high performance computing systems under uncertainty. In High Performance Computing, volume 485 of CCIS, pages 143-157. Springer

Problem formulation

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Makespan-Energy Heterogeneous Scheduling Problem
  - A set of multicore machines $P = \{m_1, \dots, m_M\}$ each having $NC(m_i)$ cores with a processing speed $S(m_i)$
  - A set of tasks $T = \{t_1, \dots, t_N\}$ each arriving at time $ARR(t_i)$
  - An *execution time* function $ET: T \times P \to \mathbb{R}^+$
  - An *energy consumption* function $EC: T \times P \to \mathbb{R}^+$
  - An *idle energy consumption* function $EC_{IDLE}: P \to \mathbb{R}^+$
  - An *execution time error* function $\Delta_{ET}: T \times P \to \mathbb{R}^+$
  - An *energy consumption error* function $\Delta_{EC}: T \times P \to \mathbb{R}$

known

unknown

- Objective
  - Minimize makespan and energy consumption

Robustness of energy-aware schedulers

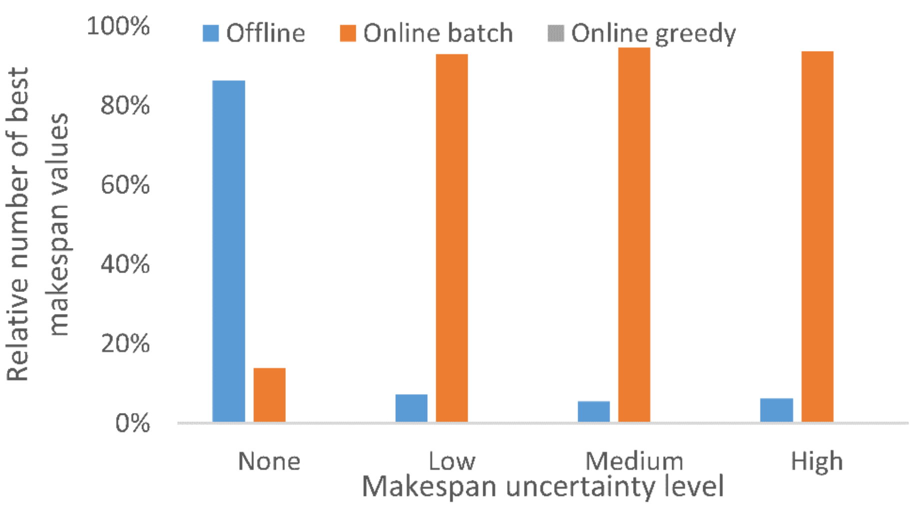
Execution time and energy uncertainty model

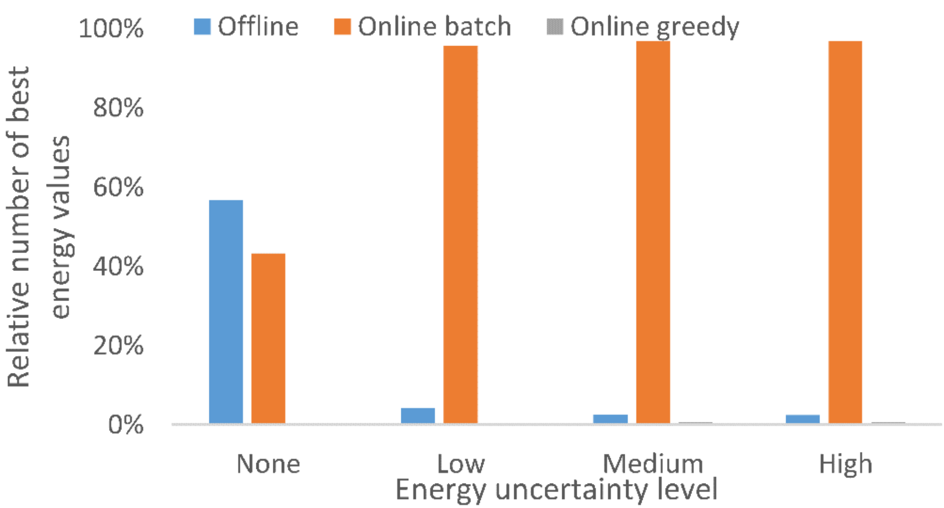
UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Empirical execution time model
  - Model built with data from three real-world data centers
- Empirical energy consumption model
  - PDU and poll and log on HP Proliant DL385 G7 24 cores, 24 GB
  - Three well-known benchmarks: single loop, LINPACK, and FFT
- Problem instances
  - Workloads of 1024 tasks
  - Scenarios with 8—16 machines (131—262 cores)
  - Low, medium, and high uncertainty levels
  - A total number of 800 problem instances
- Well-known scheduling approaches: online, batch, offline
- Online greedy algorithms: Min, MIN
- Batch/offline algorithms: MaxMin, MaxMIN, SuffMIN

Robustness of energy-aware schedulers

Results and discussion





- Offline approach: best results with no uncertainty
- Online batch approach: best results with low—high uncertainty

Robustness of energy-aware schedulers

Summary of main contributions

- Energy-aware scheduling problem considering uncertain execution times and energy consumption
- Model for workloads of tasks
  - Based on real-world data centers
- Set of realistic problem instances
- Study the robustness of greedy strategies
  - Online, batch, offline

Final conclusions

- Address energy efficient scheduling in modern data centers
- Comprehensive survey of related works
- Accurate models for energy-efficient data centers
- Diverse set of realistic problem instances
- Single data center
  - Simultaneously considering QoS, cooling, and renewable energy
  - Design and evaluate accurate multiobjective schedulers
- Federation of data centers
  - Simultaneously considering QoS, multicore, and many jobs
  - Design and evaluate accurate single- and multi-objective schedulers
- Robustness of scheduling strategies
  - Study the robustness of greedy strategies in real-world scenarios

Future work

- Integrate the proposed problem formulations
  - Cooling components and renewable energy sources
  - Federation of heterogeneous data centers
- Incorporate uncertainty to the proposed formulations
  - Extend by considering renewable energy and networking uncertainty
- Consider other renewable energy sources and applications
  - Such as wind and waste heat recycling

Thanks!

Questions?